

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Patent Application of:  
Yukio Koyanagi

Application No.: Not Yet Assigned

Confirmation No.:

Filed: Concurrently Herewith

Art Unit: N/A

For: DATA INTERPOLATING DEVICE AND  
METHOD, SAMPLING FUNCTION  
GENERATING DEVICE, DATA  
INTERPOLATING PROGRAM, AND  
RECORDED MEDIUM

Examiner: Not Yet Assigned

**CLAIM FOR PRIORITY AND SUBMISSION OF DOCUMENTS**

**MS Patent Application**  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**December 2, 2003**

Dear Sir:

Applicant hereby claims priority under 35 U.S.C. 119 based on the following prior foreign application filed in the following foreign country on the date indicated:

Country	Application No.	Date
Japan	2001-173883	June 8, 2001

In support of this claim, a certified copy of the said original foreign application is filed herewith.

Respectfully submitted,

By Larry J. Hume  
Larry J. Hume

Registration No.: 44,163  
CONNOLLY BOVE LODGE & HUTZ LLP  
1990 M Street, N.W., Suite 800  
Washington, DC 20036-3425  
(202) 331-7111  
(202) 293-6229 (Fax)  
Attorney for Applicant

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日      2001年 6月 8日  
Date of Application:

出願番号      特願2001-173883  
Application Number:

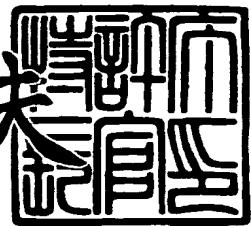
[ST. 10/C] : [JP2001-173883]

出願人      有限会社ニューロソリューション  
Applicant(s):

2003年10月 2日

特許庁長官  
Commissioner,  
Japan Patent Office

今井康夫



【書類名】 特許願

【整理番号】 13NS1332

【提出日】 平成13年 6月 8日

【あて先】 特許庁長官殿

【国際特許分類】 H03M 01/00

【発明者】

【住所又は居所】 埼玉県さいたま市中尾409-1-D115

【氏名】 小柳 裕喜生

【特許出願人】

【識別番号】 595016543

【氏名又は名称】 酒井 康江

【代理人】

【識別番号】 100105784

【弁理士】

【氏名又は名称】 橘 和之

【電話番号】 0492-49-5122

【手数料の表示】

【予納台帳番号】 070162

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0013545

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 データ補間装置および方法、標本化関数生成装置、データ補間プログラム、記録媒体

【特許請求の範囲】

【請求項1】 順次入力される離散データを順次遅延させる数段の遅延回路と、

上記数段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記順次入力される離散データに対する補間データを求める乗加算回路とを備えたことを特徴とするデータ補間装置。

【請求項2】 上記数段の遅延回路と上記乗加算回路とを1組のオーバーサンプリング回路として、複数のオーバーサンプリング回路をカスケード接続したことを特徴とする請求項1に記載のデータ補間装置。

【請求項3】 順次入力される離散データを順次遅延させる3段の遅延回路と、

上記3段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記順次入力される離散データに対する補間データを求める乗加算回路とを備えたことを特徴とするデータ補間装置。

【請求項4】 上記乗加算回路は、1段目の遅延回路の出力データを-1倍する第1の乗算器と、

2段目の遅延回路の出力データを8倍する第2の乗算器と、

3段目の遅延回路の出力データを-1倍する第3の乗算器と、

上記1段目の遅延回路の出力データと上記第1の乗算器の出力データとを選択的に切り替えて出力する第1の切替回路と、

上記3段目の遅延回路の出力データと上記第3の乗算器の出力データとを選択的に切り替えて出力する第2の切替回路と、

上記第2の乗算器の出力データ、上記第1の切替回路の出力データおよび上記第2の切替回路の出力データを加算する加算器とを備えることを特徴とする請求

項3に記載のデータ補間装置。

【請求項5】 上記乗加算回路は、1段目の遅延回路の出力データを-1倍する第1の乗算器と、2段目の遅延回路の出力データを8倍する第2の乗算器と、上記第1の乗算器の出力データ、上記第2の乗算器の出力データおよび3段目の遅延回路の出力データを加算する加算器とから成る第1の乗加算回路と、

上記2段目の遅延回路の出力データを8倍する第3の乗算器と、上記3段目の遅延回路の出力データを-1倍する第4の乗算器と、上記第3の乗算器の出力データ、上記第4の乗算器の出力データおよび上記1段目の遅延回路の出力データを加算する加算器とから成る第2の乗加算回路と、

上記第1の乗加算回路の出力データと上記第2の乗加算回路の出力データとを選択的に切り替えて出力する切替回路とを備えることを特徴とする請求項3に記載のデータ補間装置。

【請求項6】 上記乗加算回路は、1段目の遅延回路の出力データを-1倍する第1の乗算器と、

2段目の遅延回路の出力データを8倍する第2の乗算器と、

3段目の遅延回路の出力データを-1倍する第3の乗算器と、

上記第1の乗算器の出力データ、上記第2の乗算器の出力データおよび上記3段目の遅延回路の出力データを加算する第1の加算器と、

上記第2の乗算器の出力データ、上記第3の乗算器の出力データおよび上記1段目の遅延回路の出力データを加算する第2の加算器と、

上記第1の加算器の出力データと上記第2の加算器の出力データとを選択的に切り替えて出力する切替回路とを備えることを特徴とする請求項3に記載のデータ補間装置。

【請求項7】 上記3段の遅延回路と上記乗加算回路とを1組のオーバーサンプリング回路として、複数のオーバーサンプリング回路をカスケード接続したことを特徴とする請求項3に記載のデータ補間装置。

【請求項8】 順次入力される離散データを順次遅延させる数段の遅延回路と、

上記数段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、

デジタル基本関数の値に応じた重み付け加算をすることにより、上記順次入力される離散データに対する補間データを求める乗加算回路と、

上記乗加算回路から出力される隣接する補間データどうしの平均データを求める平均化回路とを備えたことを特徴とするデータ補間装置。

**【請求項 9】** 上記数段の遅延回路と上記乗加算回路と上記平均化回路とを1組のオーバーサンプリング回路として、複数のオーバーサンプリング回路をカスケード接続したことを特徴とする請求項8に記載のデータ補間装置。

**【請求項 10】** 順次入力される離散データを順次遅延させる4段の遅延回路と、

上記4段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記順次入力される離散データに対する補間データを求める乗加算回路とを備えたことを特徴とするデータ補間装置。

**【請求項 11】** 上記乗加算回路は、1段目の遅延回路の出力データを-1倍する第1の乗算器と、

2段目の遅延回路の出力データを9倍する第2の乗算器と、

3段目の遅延回路の出力データを9倍する第3の乗算器と、

4段目の遅延回路の出力データを-1倍する第4の乗算器と、

上記第1～第4の乗算器の出力データを加算する加算器と、

上記加算器の出力データと上記1段目の遅延回路に入力される離散データとを選択的に切り替えて出力する切替回路とを備えることを特徴とする請求項10に記載のデータ補間装置。

**【請求項 12】** 上記乗加算回路は、1段目の遅延回路の出力データと4段目の遅延回路の出力データとを加算する第1の加算器と、

2段目の遅延回路の出力データと3段目の遅延回路の出力データとを加算する第2の加算器と、

上記第1の加算器の出力データを-1倍する第1の乗算器と、

上記第2の加算器の出力データを9倍する第2の乗算器と、

上記第1の加算器の出力データと上記第2の加算器の出力データとを加算する

第3の加算器と、

上記第3の加算器の出力データと上記1段目の遅延回路に入力される離散データとを選択的に切り替えて出力する切替回路とを備えることを特徴とする請求項10に記載のデータ補間装置。

**【請求項13】** 上記4段の遅延回路と上記乗加算回路とを1組のオーバーサンプリング回路として、複数のオーバーサンプリング回路をカスケード接続したことを特徴とする請求項10に記載のデータ補間装置。

**【請求項14】** 順次入力される離散データを順次遅延させる5段の遅延回路と、

上記5段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記順次入力される離散データに対する補間データを求める乗加算回路とを備えたことを特徴とするデータ補間装置。

**【請求項15】** 上記乗加算回路は、1段目の遅延回路の出力データを-1倍する第1の乗算器と、2段目の遅延回路の出力データを9倍する第2の乗算器と、3段目の遅延回路の出力データを25倍する第3の乗算器と、4段目の遅延回路の出力データを-1倍する第4の乗算器と、上記第1～第4の乗算器の出力データを加算する加算器とから成る第1の乗加算回路と、

上記2段目の遅延回路の出力データを-1倍する第5の乗算器と、上記4段目の遅延回路の出力データを9倍する第6の乗算器と、上記第5の乗算器の出力データを-1倍する第7の乗算器と、上記第3の乗算器の出力データおよび上記第5～第7の乗算器の出力データを加算する加算器とから成る第2の乗加算回路と、

上記第1の乗加算回路の出力データと上記第2の乗加算回路の出力データとを選択的に切り替えて出力する切替回路とを備えることを特徴とする請求項14に記載のデータ補間装置。

**【請求項16】** 上記5段の遅延回路と上記乗加算回路とを1組のオーバーサンプリング回路として、複数のオーバーサンプリング回路をカスケード接続したことを特徴とする請求項14に記載のデータ補間装置。

【請求項 17】 順次入力される離散データについて、注目するサンプル点の離散データとその周辺数個のサンプル点の離散データを取得するデータ取得手段と、

上記データ取得手段により取得した数個の離散データに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記注目するサンプル点の離散データに対する補間データを順次求める補間手段とを備えたことを特徴とするデータ補間装置。

【請求項 18】 上記データ取得手段および上記補間手段の処理を、上記補間手段の出力データを上記データ取得手段の入力として数回繰り返し行うこととする特徴とする請求項 17 記載のデータ補間装置。

【請求項 19】 上記補間手段により求められた補間データについて、隣接する補間データどうしで平均化演算を行う平均化手段を備えたことを特徴とする請求項 17 に記載のデータ補間装置。

【請求項 20】 順次入力される離散データについて、注目するサンプル点の離散データとその周辺数個のサンプル点の離散データとに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記注目するサンプル点の離散データに対する補間データを求めるようにしたことを特徴とするデータ補間方法。

【請求項 21】 上記順次入力される離散データを、上記デジタル基本関数の値に応じた重み付け加算により求めた 2 つの補間データにそれぞれ置き換えるようにしたことを特徴とする請求項 20 に記載のデータ補間方法。

【請求項 22】 上記デジタル基本関数の値に応じた重み付け加算により求められた補間データについて、隣接する補間データどうしで平均化演算を行うようにしたことを特徴とする請求項 21 に記載のデータ補間方法。

【請求項 23】 上記デジタル基本関数の値に応じた重み付け加算により求められた補間データについて、注目するサンプル点の補間データとその周辺数個のサンプル点の補間データとに対して、上記デジタル基本関数の値に応じた重み付け加算をすることにより、上記注目するサンプル点の補間データに対する補間データを更に求めるようにしたことを特徴とする請求項 20 に記載のデータ補間

方法。

**【請求項 24】** 順次入力される離散データについて、注目する1つのサンプル点の離散データとその両隣のサンプル点の離散データとに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記1つのサンプル点の離散データから2つのサンプル点の補間データを順次求めるようにしたことを特徴とするデータ補間方法。

**【請求項 25】** 上記求められた補間データについて、隣接する補間データどうしで平均化演算を行うようにしたことを特徴とする請求項24に記載のデータ補間方法。

**【請求項 26】** 順次入力される離散データについて、注目する隣接する2つのサンプル点の離散データとその両隣のサンプル点の離散データとに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記2つのサンプル点の間を補間する補間データを順次求めるようにしたことを特徴とするデータ補間方法。

**【請求項 27】** 入力される単一のデジタルデータを順次遅延させる数段の遅延回路と、

上記数段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記単一のデジタルデータに対する補間データを求める乗加算回路とから1組のオーバーサンプリング回路を構成し、

複数のオーバーサンプリング回路をカスケード接続したことを特徴とする標準化関数生成装置。

**【請求項 28】** 請求項17～19の何れか1項に記載の各手段としてコンピュータを機能させるためのデータ補間プログラム。

**【請求項 29】** 請求項20～26の何れか1項に記載のデータ補間方法の処理手順をコンピュータに実行させるためのデータ補間プログラム。

**【請求項 30】** 請求項17～19の何れか1項に記載の各手段としてコンピュータを機能させるためのプログラムを記録したコンピュータ読み取り可能な記録媒体。

【請求項31】 請求項20～26の何れか1項に記載のデータ補間方法の処理手順をコンピュータに実行させるためのプログラムを記録したコンピュータ読み取り可能な記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明はデータ補間装置および方法、標本化関数生成装置、データ補間プログラム、記録媒体に関し、特に、音声や画像などの離散的なデータ間を補間するデータ補間方式に用いて好適なものである。

【0002】

【従来の技術】

従来、あらかじめ与えられた標本値間の値を求めるデータ補間方法として、様々な方法が提案されている。最も簡単な方法は、直線補間である。この直線補間では、例えば2つの標本値  $x_1, x_2$  の間を補間する中間値として、 $(x_1 + x_2) / 2$  で表される補間データを求める。

【0003】

また、所定の標本化関数を用いてデータ補間を行う方法も知られている。この場合のデータ補間には、通常、従来から知られている  $\sin$  関数と称される標本化関数が用いられる。

【0004】

図14は、 $\sin$  関数の説明図である。 $\sin$  関数は、ディラックのデルタ関数を逆フーリエ変換したときに現れるものであり、標本化周波数を  $f$  としたときに  $\sin(\pi f t) / (\pi f t)$  で定義される。この  $\sin$  関数は、 $t = 0$  のサンプル点のみで値が1になり、他の全てのサンプル点では値が0となる。

【0005】

図15は、図14に示した標本化関数を用いたデータ補間の説明図である。図15において、等間隔のサンプル点  $t_1, t_2, t_3, t_4$  のそれぞれにおける離散データの値を  $Y(t_1), Y(t_2), Y(t_3), Y(t_4)$  とし、例えばサンプル点  $t_2$  と  $t_3$  の間の所定位置  $t_0$  ( $t_2$  から距離  $a$  の位置) に対応し

た補間値  $y(t_0)$  を求める場合を考える。

#### 【0006】

一般に、補間値  $y(t_0)$  を標本化関数を用いて求めるには、各サンプル点  $t_1 \sim t_4$  の離散データ値  $Y(t_1) \sim Y(t_4)$  に対応した振幅を有する複数の標本化関数について、補間位置  $t_0$  における値（ $\times$ 印で示す）を夫々求め、これらを畳み込み演算によって全て加算すればよい。

#### 【0007】

これらのデータ補間は、例えば、離散的なデジタルデータを連続的なアナログ信号に変換するデジタルーアナログ変換処理や、圧縮時に離散化されたデータから補間によって連続的な元データを復元する伸長処理などに利用される。その際には、与えられた離散的なサンプル点間の値をできるだけ滑らかに補間することが要求されることが多い。

#### 【0008】

この要求に対応するために、補間処理によって擬似的にサンプリング周波数を上げる従来のデータ補間方法では、求めた各補間値をサンプルホールド回路によって保持して階段状の信号波形を生成した後に、それをローパスフィルタに通すことによって滑らかな信号を出力するようになっていた。

#### 【0009】

ところが、この方法では、出力される連続信号に、ローパスフィルタによる位相特性の劣化が生じるという問題があった。

また、上述した  $\sin c$  関数は、 $\pm\infty$  で 0 に収束する関数であるため、正確な補間値を求めようとすると、全ての離散データ値に対応した  $\sin c$  関数の値を求めて加算する必要がある。ところが実際には、処理能力や回路規模等の都合から、考慮する離散データの範囲を限定して畳み込み演算が行われていた。そのため、得られる補間値には打ち切り誤差が含まれ、正確な補間値が得られないという問題があった。

#### 【0010】

このような問題を解決するために、本出願人は、ローパスフィルタによる位相特性の劣化や打ち切り誤差がなく、歪みの少ない出力波形を得ることが可能なデ

ータ補間方式を発明し、既に特許出願した（特願平11-173245号）。この特願平11-173245号に記載の発明は、図12に示すデジタル基本関数を入力されるn個の離散データの値に応じた振幅に加工し、それらをオーバーサンプリングおよび畳み込み演算によって合成することにより、結果として図13に示すような標本化関数に基づきデータ補間を行うようにしたものである。

### 【0011】

この図13に示す標本化関数は、全域において1回微分可能であって、横軸に沿った標本位置tが有限の局所的な領域内でのみ0以外の有限な値を有し、それ以外の領域では値が全て0となる有限台の関数である。

また、図13の標本化関数は、サンプル点t3のみで極大値をとり、4つのサンプル点t1, t2, t4, t5で値が0になるという特徴を持つ関数であり、滑らかな波形の信号を得るために必要なサンプル点は全て通る。

### 【0012】

このように、図13に示す関数は、全域において1回微分可能であり、しかもサンプル点t1, t5において0に収束する有限台の標本化関数である。したがって、図14に示した従来のsin関数の代わりに、図13の標本化関数を用いて各離散データに基づく重ね合わせを行うことにより、離散データ間の値を1回微分可能な関数を用いて滑らかに補間することが可能となる。これにより、ローパスフィルタを不要とし、その位相特性の劣化を防止することができる。

### 【0013】

また、図13の標本化関数は、サンプル点t1, t5で0に収束するため、t1～t5の範囲内での離散データだけを考慮に入れればよい。したがって、ある1つの補間値を求める場合には、限られた数の離散データの値のみを考慮すればよいことになり、処理量を大幅に削減することができる。しかも、t1～t5の範囲外の各離散データについては、本来考慮すべきであるが処理量や精度等を考慮して無視しているというわけではなく、理論的に考慮する必要がないため、打ち切り誤差の発生を防止することもできる。

### 【0014】

#### 【発明が解決しようとする課題】

しかしながら、上記特願平11-173245号に記載されたデータ補間方式では、上述のように優れた特性は得ることができるものの、置み込み演算あるいは移動平均演算など、直線補間などの簡単なデータ補間方式に比べて複雑な演算を行う必要があった。

### 【0015】

そこで本発明は、全域において1回以上微分可能な有限台の標本化関数に基づく滑らかなデータ補間をより簡単な演算で行うことができるようにして、演算時間の短縮化や、その演算を行うための回路の簡素化を実現できるようにすることを目的とする。

### 【0016】

#### 【課題を解決するための手段】

本発明のデータ補間装置は、順次入力される離散データを順次遅延させる数段の遅延回路と、上記数段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記順次入力される離散データに対する補間データを求める乗加算回路とを備えたことを特徴とする。

### 【0017】

本発明の他の態様では、上記数段の遅延回路と上記乗加算回路とを1組のオーバーサンプリング回路として、複数のオーバーサンプリング回路をカスケード接続したことを特徴とする。

### 【0018】

本発明のその他の態様では、順次入力される離散データを順次遅延させる3段の遅延回路と、上記3段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記順次入力される離散データに対する補間データを求める乗加算回路とを備えたことを特徴とする。

### 【0019】

本発明のその他の態様では、上記3段の遅延回路と上記乗加算回路とを1組のオーバーサンプリング回路として、複数のオーバーサンプリング回路をカスケー

ド接続したことを特徴とする。

#### 【0020】

本発明のその他の態様では、順次入力される離散データを順次遅延させる数段の遅延回路と、上記数段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記順次入力される離散データに対する補間データを求める乗加算回路と、上記乗加算回路から出力される隣接する補間データどうしの平均データを求める平均化回路とを備えたことを特徴とする。

#### 【0021】

本発明のその他の態様では、上記数段の遅延回路と上記乗加算回路と上記平均化回路とを1組のオーバーサンプリング回路として、複数のオーバーサンプリング回路をカスケード接続したことを特徴とする。

#### 【0022】

本発明のその他の態様では、順次入力される離散データを順次遅延させる4段の遅延回路と、上記4段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記順次入力される離散データに対する補間データを求める乗加算回路とを備えたことを特徴とする。

#### 【0023】

本発明のその他の態様では、上記4段の遅延回路と上記乗加算回路とを1組のオーバーサンプリング回路として、複数のオーバーサンプリング回路をカスケード接続したことを特徴とする。

#### 【0024】

本発明のその他の態様では、順次入力される離散データを順次遅延させる5段の遅延回路と、上記5段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記順次入力される離散データに対する補間データを求める乗加算回路とを備えたことを特徴とする。

#### 【0025】

本発明のその他の態様では、上記5段の遅延回路と上記乗加算回路とを1組のオーバーサンプリング回路として、複数のオーバーサンプリング回路をカスケード接続したことを特徴とする。

#### 【0026】

本発明のその他の態様では、順次入力される離散データについて、注目するサンプル点の離散データとその周辺数個のサンプル点の離散データを取得するデータ取得手段と、上記データ取得手段により取得した数個の離散データに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記注目するサンプル点の離散データに対する補間データを順次求める補間手段とを備えたことを特徴とする。

#### 【0027】

本発明のその他の態様では、上記データ取得手段および上記補間手段の処理を、上記補間手段の出力データを上記データ取得手段の入力として数回繰り返し行うことを特徴とする。

#### 【0028】

本発明のその他の態様では、上記補間手段により求められた補間データについて、隣接する補間データどうしで平均化演算を行う平均化手段を備えたことを特徴とする。

#### 【0029】

また、本発明のデータ補間方法は、順次入力される離散データについて、注目するサンプル点の離散データとその周辺数個のサンプル点の離散データとに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記注目するサンプル点の離散データに対する補間データを求めるようにしたことを特徴とする。

#### 【0030】

本発明の他の態様では、上記デジタル基本関数の値に応じた重み付け加算により求められた補間データについて、隣接する補間データどうしで平均化演算を行うようにしたことを特徴とする。

#### 【0031】

本発明のその他の態様では、上記デジタル基本関数の値に応じた重み付け加算により求められた補間データについて、注目するサンプル点の補間データとその周辺数個のサンプル点の補間データとに対して、上記デジタル基本関数の値に応じた重み付け加算をすることにより、上記注目するサンプル点の補間データに対する補間データを更に求めるようにしたことを特徴とする。

#### 【0032】

本発明のその他の態様では、順次入力される離散データについて、注目する1つのサンプル点の離散データとその両隣のサンプル点の離散データとに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記1つのサンプル点の離散データから2つのサンプル点の補間データを順次求めるようにしたことを特徴とする。

#### 【0033】

本発明のその他の態様では、順次入力される離散データについて、注目する隣接する2つのサンプル点の離散データとその両隣のサンプル点の離散データとに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記2つのサンプル点の間を補間する補間データを順次求めるようにしたことを特徴とする。

#### 【0034】

また、本発明の標本化関数生成装置は、入力される单一のデジタルデータを順次遅延させる数段の遅延回路と、上記数段の遅延回路の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数の値に応じた重み付け加算をすることにより、上記单一のデジタルデータに対する補間データを求める乗加算回路とから1組のオーバーサンプリング回路を構成し、複数のオーバーサンプリング回路をカスケード接続したことを特徴とする。

#### 【0035】

また、本発明のデータ補間プログラムは、請求項17～19の何れか1項に記載の各手段としてコンピュータを機能させるためのプログラム、あるいは、請求項20～26の何れか1項に記載のデータ補間方法の処理手順をコンピュータに実行させるためのプログラムである。

### 【0036】

また、本発明のコンピュータ読み取り可能な記録媒体は、請求項17～19の何れか1項に記載の各手段としてコンピュータを機能させるためのプログラム、あるいは、請求項20～26の何れか1項に記載のデータ補間方法の処理手順をコンピュータに実行させるためのプログラムを記録したことを特徴とする。

### 【0037】

#### 【発明の実施の形態】

##### (第1の実施形態)

以下、本発明の第1の実施形態を図面に基づいて説明する。

図1は、第1の実施形態によるデータ補間装置の構成例を示す図である。図2は、第1の実施形態によるデータ補間の原理を説明するための図である。まず、図2を用いて本実施形態によるデータ補間の原理について説明する。

### 【0038】

図2に示すデータ構造において、横軸のA,B,C,…は1クロック毎に順次入力される各サンプル点の離散データを示し、そのデータ値がデジタル基本関数に対する係数となる。また、縦軸のa,b,c,…は、デジタル基本関数により加工された離散データの中心位置を示す。

### 【0039】

ここで用いるデジタル基本関数は、オーバーサンプリングによるデータ補間を行う際に使用する標本化関数の基本となるものであり、図10に示されるものである。このデジタル基本関数は、1クロック毎にデータ値を-1, 1, 8, 8, 1, -1と変化させて作成したものである。

### 【0040】

図2に示すように、離散データAに対するデジタル基本関数は、最初のクロック位置を先頭として各関数值(-1, 1, 8, 8, 1, -1)を1クロックずつずらして与えられる。また、次の離散データBに対するデジタル基本関数は、最初のクロック位置より2クロック後のクロック位置を先頭として各関数值を1クロックずつずらして与えられる。同様に、順次入力される各離散データC,D,E,F,…に対するデジタル基本関数は、更に2クロックずつ遅れたクロック位置をそれ

ぞれ先頭として、各関数値を1クロックずつずらして与えられる。

#### 【0041】

このようなデータ構造に基づいて、1つの離散データBから2つの補間データB<sub>1</sub>, B<sub>2</sub>を生成する場合を例に挙げて説明する。ここでは、デジタル基本関数により加工された離散データBの中心位置bを挟む2行のデータをそれぞれb<sub>1</sub>, b<sub>2</sub>とし、これらのデータb<sub>1</sub>, b<sub>2</sub>に対して畳み込み演算を行うことを考える。この場合、2つの隣接するデータ値b<sub>1</sub>, b<sub>2</sub>は、それぞれ

$$b_1 = A + 8B - C \quad \cdots (1)$$

$$b_2 = -A + 8B + C \quad \cdots (2)$$

と表される。

#### 【0042】

上記式(1)(2)より、

$$b_1 + b_2 = 16B \quad \cdots (3)$$

と表され、さらに、

$$B = (b_1 / 8 + b_2 / 8) / 2 \quad \cdots (4)$$

のように変形される。

#### 【0043】

上記式(4)から、離散データBは、2つのデータb<sub>1</sub>/8, b<sub>2</sub>/8の中間値に相当すると言える。これを逆に考えることで、1つの離散データBは、2つの補間データB<sub>1</sub>, B<sub>2</sub>に置き換えることができ、それらは、

$$B_1 = (A + 8B - C) / 8 \quad \cdots (5)$$

$$B_2 = (-A + 8B + C) / 8 \quad \cdots (6)$$

のように表される。

#### 【0044】

また、離散データBについて式(4)に示される関係が他の離散データC, D, E, F, …についても同様に成り立つことから、

$$C = (c_1 / 8 + c_2 / 8) / 2$$

$$D = (d_1 / 8 + d_2 / 8) / 2$$

$$E = (e_1 / 8 + e_2 / 8) / 2$$

$$F = (f_1/8 + f_2/8)/2$$

.....

となる。

#### 【0045】

したがって、次式に示すように、1つの離散データCは2つの補間データC1, C2に、1つの離散データDは2つの補間データD1, D2に、1つの離散データEは2つの補間データE1, E2に、1つの離散データFは2つの補間データF1, F2にそれぞれ置き換えられる。

$$C \rightarrow C1 = (B + 8C - D)/8, C2 = (-B + 8C + D)/8$$

$$D \rightarrow D1 = (C + 8D - E)/8, D2 = (-C + 8D + E)/8$$

$$E \rightarrow E1 = (D + 8E - F)/8, E2 = (-D + 8E + F)/8$$

$$F \rightarrow F1 = (E + 8F - G)/8, F2 = (-E + 8F + G)/8$$

.....

#### 【0046】

以上のように、あるサンプル点の離散データ（例えばB）に関して補間値を求める際には、その離散データを8倍した値と、前後のサンプル点の離散データを+1倍、-1倍した値とを加算し、その加算値を8で割ることによって第1の補間値（例えばB1）を得る。また、対象の離散データを8倍した値と、前後のサンプル点の離散データを-1倍、+1倍した値とを加算し、その加算値を8で割ることによって第2の補間値（例えばB2）を得る。これを各サンプル点について行うことにより、元データを2倍にオーバーサンプルする。

#### 【0047】

また、2倍にオーバーサンプルされた補間データをもとに同様の補間処理を行うことにより、元データを4倍にオーバーサンプルすることが可能である。さらに、4倍にオーバーサンプルされた補間データをもとに同様の補間処理を行うことにより、元データを8倍にオーバーサンプルすることが可能である。理論的には、同様の補間処理をn回繰り返し行うことにより、元データを2^n倍にオーバーサンプルすることが可能である。

#### 【0048】

次に、図1を用いて、上述の補間動作を実現する第1の実施形態によるデータ補間装置の構成について説明する。

図1に示すデータ補間装置は、2倍のオーバーサンプリングを行うための回路を3段カスケード接続した構成になっており、1段目～3段目の回路構成を符号の添数字（-1～-3）により区別している。また、添数字のみが異なり主符号が同じ回路構成は互いに同一の機能を有するものであることを示している。

#### 【0049】

1段目のオーバーサンプリング回路は、基準周波数（例えば44.1KHz）のクロックCKに従って動作する。3段のD型フリップフロップ $1_{-1}$ ,  $2_{-1}$ ,  $3_{-1}$ は、順次入力される離散データ（例えば16ビット）を基準周波数の1クロックCKずつ順に遅延させる。これらのD型フリップフロップ $1_{-1}$ ,  $2_{-1}$ ,  $3_{-1}$ は、本発明による3段の遅延回路に相当する。

#### 【0050】

1段目のD型フリップフロップ $1_{-1}$ の出力タップから取り出されるデータは、例えば式(5)(6)に示す第1項を形成し、2段目のD型フリップフロップ $2_{-1}$ の出力タップから取り出されるデータは第2項を形成し、3段目のD型フリップフロップ $3_{-1}$ の出力タップから取り出されるデータは第3項を形成する。

#### 【0051】

1段目のD型フリップフロップ $1_{-1}$ の出力タップから取り出されたデータは、-1倍の乗算器 $4_{-1}$ （請求項3の第1の乗算器に相当）を介してANDゲート $5_{-1}$ の一方の入力端に入力されるとともに、乗算器を介すことなく(+1倍に相当)ANDゲート $6_{-1}$ の一方の入力端に入力される。ANDゲート $5_{-1}$ の他方の入力端には、インバータ $7_{-1}$ を通過した反転クロックCKが入力される。また、ANDゲート $6_{-1}$ の他方の入力端には、クロックCKが入力される。

#### 【0052】

上記2つのANDゲート $5_{-1}$ ,  $6_{-1}$ の出力データは、ORゲート $8_{-1}$ を介して出力される。これにより、クロックCKが“H”の期間中は+1倍された離散データがORゲート $8_{-1}$ から出力される。また、クロックCKが“L”の期間中は-1倍された離散データがORゲート $8_{-1}$ から出力される。すなわち、クロック

CKが“H”の期間中に上記式(5)に示す第1項が求められ、“L”の期間中に上記式(6)に示す第1項が求められる。2つのANDゲート $5_{-1}$ ,  $6_{-1}$ , インバータ $7_{-1}$ およびORゲート $8_{-1}$ により請求項3の第1の切替回路が構成される。

#### 【0053】

2段目のD型フリップフロップ $2_{-1}$ の出力タップから取り出されたデータは、+8倍の乗算器 $9_{-1}$ （請求項3の第2の乗算器に相当）を介して出力される。上記式(5)(6)に示すように、第2項については両式において±の符号が変わることはないので、第1項のようなクロックCKによる符号切替のための回路は不要である。

#### 【0054】

また、3段目のD型フリップフロップ $3_{-1}$ の出力タップから取り出されたデータは、-1倍の乗算器 $10_{-1}$ （請求項3の第3の乗算器に相当）を介してANDゲート $11_{-1}$ の一方の入力端に入力されるとともに、乗算器を介すことなく(+1倍に相当) ANDゲート $12_{-1}$ の一方の入力端に入力される。ANDゲート $11_{-1}$ の他方の入力端には、クロックCKが入力される。また、ANDゲート $12_{-1}$ の他方の入力端には、インバータ $13_{-1}$ を通過した反転クロックCKが入力される。

#### 【0055】

上記2つのANDゲート $11_{-1}$ ,  $12_{-1}$ の出力データは、ORゲート $14_{-1}$ を介して出力される。これにより、クロックCKが“H”的期間中は-1倍された離散データがORゲート $14_{-1}$ から出力される。また、クロックCKが“L”的期間中は+1倍された離散データがORゲート $14_{-1}$ から出力される。すなわち、クロックCKが“H”的期間中に上記式(5)に示す第3項が求められ、“L”的期間中に上記式(6)に示す第3項が求められる。2つのANDゲート $11_{-1}$ ,  $12_{-1}$ , インバータ $13_{-1}$ およびORゲート $14_{-1}$ により請求項3の第2の切替回路が構成される。

#### 【0056】

上記ORゲート $8_{-1}$ の出力データ、8倍乗算器 $9_{-1}$ の出力データ、およびORゲート $14_{-1}$ の出力データは、2つの加算器 $15_{-1}$ ,  $16_{-1}$ （請求項3の加算器

に相当) によって全て加算される。これにより、クロックCKが“H”の期間中に上記式(5)に示す演算結果が加算器16-1より出力され、クロックCKが“L”の期間中に上記式(6)に示す演算結果が加算器16-1より出力される。

#### 【0057】

以上のような回路に対して、図2に示すような離散データA,B,C,…を順に入力することにより、加算器16-1からは、A,B1,B2,C1,C2,…のように2倍にオーバーサンプリングされた補間データが出力される。なお、式(5)(6)ではデータ値が全体として $1/8$ 倍されているが、そのための回路が1段目のオーバーサンプリング回路には設けられていない。図1の回路では、3段分をまとめて最終段の乗算器20で $1/512$  ( $1/8^3$ ) 倍している。

#### 【0058】

図1に示すデータ補間装置では、例えば1つの離散データBから2つの補間データB1, B2を生成する際に、両補間データで共通している式(5)(6)の第2項については、1つの8倍乗算器9-1を共有してデータ形成するようにしている。このように、2つの補間データを生成する際に共有できる回路はできる限り共有化することにより、全体としての回路構成を簡素化している。

#### 【0059】

上記加算器16-1の出力段には、D型フリップフロップ17-1と、加算器18-1と、 $1/2$ 倍乗算器19-1とから成る平均化回路が設けられている。D型フリップフロップ17-1は、基準周波数の2倍の周波数のクロック2CKに従って、加算器16-1の出力データを1クロック分遅延させる。加算器18-1は、加算器16-1の出力データとD型フリップフロップ17-1の出力データとを加算する。 $1/2$ 倍乗算器19-1は、加算器18-1の出力データを $1/2$ 倍する。

#### 【0060】

このように平均化回路を設けているのは、補間処理によって生じたクロックの相対位置のずれを元に戻すためである。例えば、加算器16-1からB1, B2, C1, C2, …のように補間データが出力されている場合、平均化回路では、 $(B1+B2)/2$ ,  $(B2+C1)/2$ ,  $(C1+C2)/2$ , …の演算を順に実行する。

#### 【0061】

この平均化演算の実行結果は、

$$(B1+B2)/2 = \{(A+8B-C)/8\} + \{(-A+8B+C)/8\}/2 = B$$

$$(B2+C1)/2 = \{(-A+8B+C)/8\} + \{(B+8C-D)/8\}/2$$

$$= (-A+9B+9C-D)/16 \quad \cdots (7)$$

$$(C1+C2)/2 = \{(B+8C-D)/8\} + \{(-B+8C+D)/8\}/2 = C$$

.....

となり、クロックの相対位置のずれが元の正しい位置に修正される。これによつて、図6を用いて後述するように、得られる標本化関数の周波数特性を良好にすることが可能となる。

#### 【0062】

上記平均化回路を構成する  $1/2$  倍乗算器  $19_{-1}$  の出力データは、2段目のオーバーサンプリング回路のD型フリップフロップ  $1_{-2}$  に入力され、2段目の  $2$  倍オーバーサンプリングが実行される。2段目のオーバーサンプリング回路は、基準周波数の  $2$  倍の周波数のクロック  $2CK$  に従って動作すること以外は、1段目のオーバーサンプリング回路と同様である。

#### 【0063】

2段目のオーバーサンプリング回路の最終段に設けられた  $1/2$  倍乗算器  $19_{-2}$  の出力データは、3段目のオーバーサンプリング回路のD型フリップフロップ  $1_{-3}$  に入力され、3段目の  $2$  倍オーバーサンプリングが実行される。3段目のオーバーサンプリング回路は、基準周波数の  $4$  倍の周波数のクロック  $4CK$  に従つて動作すること以外は、1段目のオーバーサンプリング回路と同様である。

#### 【0064】

図3は、上記図1に示したデータ補間装置にユニットパルスのデータを入力した場合に得られるオーバーサンプル結果を示す特性図である。図3において、①はユニットパルスの入力データ、②は  $2$  倍オーバーサンプルデータ、③は  $4$  倍オーバーサンプルデータ、④は  $8$  倍オーバーサンプルデータを示す。④に示すオーバーサンプルデータの波形関数は、図13に示した特願平11-173245号に記載の畳み込み演算により得られる標本化関数とほぼ同じ波形を有している。

#### 【0065】

参考までに、図3の④に示す波形関数と、図13に示した波形関数とを比較のために並べて図4に示す。図4において、実線のグラフIは図3の④に示す波形関数、点線のグラフIIは図13に示す波形関数である。これから分かるように、両者の波形にはほとんど差はなく、有限の標本位置において0に収束する有限台の標本化関数である。しかも、図3の④に示す関数は、全域において1回以上微分可能である。

#### 【0066】

したがって、本実施形態のデータ補間処理を適用した場合は、結果として図3の④に示す標本化関数を用いて各離散データに基づく重ね合わせを行うこととなり、離散データ間の値を1回以上微分可能な関数を用いてより滑らかに補間することが可能となる。これにより、ローパスフィルタを不要とし、その位相特性の劣化を防止することができる。

#### 【0067】

また、ある1つの離散データに関して補間値を求める場合には、限られた数（上記式(5)(6)で示されるように3個）の離散データの値のみを考慮すればよいことになり、処理量を大幅に削減することができる。しかも、有限台の範囲外の各離散データについては、本来考慮すべきであるが処理量や精度等を考慮して無視しているというわけではなく、理論的に考慮する必要がないため、打ち切り誤差の発生を防止することもできる。

#### 【0068】

図5は、図4に示した標本化関数の一部拡大図である。図5に示すように、特願平11-173245号により得られる標本化関数IIは、サンプル点  $t = 7$  の位置にて強制的に0に収束する形になっているが、本実施形態により得られる標本化関数Iは、そのような強制的な打ち切りがない。したがって、本実施形態によれば、離散データ間をより滑らかに補間することができる。

#### 【0069】

図6は、図4に示した標本化関数の周波数特性を示す図である。図6において、実線のグラフIは本実施形態により得られる標本化関数の周波数特性を示し、点線のグラフIIは特願平11-173245号により得られる標本化関数の周波

数特性を示す。この図6から分かるように、両方とも極めて良好な周波数特性が得られている。これによれば、いわゆる折り返しノイズが出るところにちょうどノッチフィルタが入ったような形になっており、折り返しノイズの発生を効果的に抑制することができる。

### 【0070】

以上詳しく説明したように、第1の実施形態によれば、全域において1回以上微分可能な有限台の標本化関数を、上記式(5)(6)のような極めて簡単な四則演算のみで得ることができ、当該標本化関数に基づくデータ補間を簡単な演算で行うことができる。この式(5)(6)によれば、ある注目するサンプル点の離散データについて補間データを求める際には、その両隣のサンプル点の離散データのみを考慮に入れれば良い。したがって、本実施形態によれば、ローパスフィルタの位相歪みや打ち切り誤差のない良好な補間データを得ることができることはもちろん、そのための演算時間を短縮化することができるとともに、その演算回路を簡素化することができる。

### 【0071】

なお、上記図1に示したデータ補間回路は、本発明のデータ補間手法を実現するための回路構成の一例であって、これに限定されるものではない。

例えば、3つのD型フリップフロップの出力タップから取り出されるそれぞれのデータを-1倍、8倍、+1倍して加算する第1の乗加算回路と、3つのD型フリップフロップの出力タップから取り出されるそれぞれのデータを+1倍、8倍、-1倍して加算する第2の乗加算回路と別個に設けるようにしても良い。

### 【0072】

図7は、この場合の回路構成例を示す図である。なお、ここでは2倍のオーバーサンプルを行う回路のみを示しているが、同様の回路をn個カスケード接続することにより、 $2^n$ 倍のオーバーサンプルを行うことができるのは、図1の場合と同様である。また、ここでは平均化回路の図示を省略している。

### 【0073】

図7において、3段のD型フリップフロップ21, 22, 23は、順次入力される離散データを1クロックCKずつ順に遅延させる。第1の乗加算回路は、1

段目のD型フリップフロップ21の出力データを-1倍する第1の乗算器24と、2段目のD型フリップフロップ22の出力データを8倍する第2の乗算器25と、第1の乗算器24の出力データ、第2の乗算器25の出力データおよび3段目のD型フリップフロップ23の出力データを加算する加算器26, 27とを備える。

#### 【0074】

また、第2の乗加算回路は、2段目のD型フリップフロップ22の出力データを8倍する第3の乗算器28と、3段目のD型フリップフロップ23の出力データを-1倍する第4の乗算器29と、第3の乗算器28の出力データ、第4の乗算器29の出力データおよび1段目のD型フリップフロップ21の出力データを加算する加算器30, 31とを備える。

#### 【0075】

第1の乗加算回路（加算器27）から出力されたデータは、ANDゲート32の一方の入力端に入力される。ANDゲート32の他方の入力端には、インバータ33を通過した反転クロックCKが入力される。また、第2の乗加算回路（加算器31）から出力されたデータは、ANDゲート34の一方の入力端に入力される。ANDゲート34の他方の入力端には、クロックCKが入力される。

#### 【0076】

上記2つのANDゲート32, 34の出力データは、ORゲート35を介して出力される。これにより、クロックCKが“H”の期間中は、上記式(5)に示すような演算結果がORゲート35より出力され、クロックCKが“L”の期間中は、上記式(6)に示すような演算結果がORゲート35より出力される。

#### 【0077】

また、図8に示すように、図7では2つ用いていた8倍乗算器を1つの乗算器で兼用するようにしても良い。なお、図8でも2倍のオーバーサンプルを行う回路のみを示しているが、同様の回路をn個カスケード接続することにより、 $2^n$ 倍のオーバーサンプルを行うことができる。また、ここでも平均化回路の図示は省略している。

#### 【0078】

図8に示すデータ補間装置は、順次入力される離散データを1クロックCKずつ順に遅延させる3段のD型フリップフロップ41, 42, 43を備えている。また、1段目のD型フリップフロップ41の出力データを-1倍する第1の乗算器44と、2段目のD型フリップフロップ42の出力データを8倍する第2の乗算器45と、3段目のD型フリップフロップ43の出力データを-1倍する第3の乗算器46とを備えている。

#### 【0079】

さらに、第1の乗算器44の出力データ、第2の乗算器45の出力データおよび3段目のD型フリップフロップ43の出力データを加算する第1の加算器47, 48と、第2の乗算器45の出力データ、第3の乗算器46の出力データおよび1段目のD型フリップフロップ41の出力データを加算する第2の加算器49, 50とを備えている。

#### 【0080】

加算器48から出力されたデータは、ANDゲート51の一方の入力端に入力される。ANDゲート51の他方の入力端には、インバータ52を通過した反転クロックCKが入力される。また、加算器50から出力されたデータは、ANDゲート53の一方の入力端に入力される。ANDゲート53の他方の入力端には、クロックCKが入力される。

#### 【0081】

上記2つのANDゲート51, 53の出力データは、ORゲート54を介して出力される。これにより、クロックCKが“H”の期間中は、上記式(5)に示すような演算結果がORゲート54より出力され、クロックCKが“L”の期間中は、上記式(6)に示すような演算結果がORゲート54より出力される。

#### 【0082】

また、上記実施形態では、上記式(5)(6)のような演算を行うことによってデータ補間を行い、求めた補間データに対して更に平均化演算を行うことによってクロックの相対位置のずれを補正するようにしていた。これに対して、上記式(7)に示すような演算を直接行うことによって、平均化演算を省略できるようにしても良い。

## 【0083】

上記式(7)に示すような演算を直接行う場合は、順次入力される離散データについて、注目する隣接する2つのサンプル点の離散データとその両隣のサンプル点の離散データとに対して、デジタル基本関数の値に応じた重み付け加算することにより、上記2つのサンプル点の離散データの間を補間する補間データを順次求めることになる。例えば、注目する2つの離散データがB,Cの場合は、その間の補間データ  $(B+C)/2$  は、次式のように求められる。

$$(B+C)/2 = (-A+9B+9C-D)/16 \quad \cdots (8)$$

## 【0084】

図9は、この場合の回路構成例を示す図である。なお、ここでも2倍のオーバーサンプルを行う回路のみを示しているが、同様の回路をn個カスケード接続することにより、 $2^n$ 倍のオーバーサンプルを行うことが可能である。

## 【0085】

図9に示すデータ補間装置は、順次入力される離散データを1クロックCKずつ順に遅延させる4段のD型フリップフロップ61, 62, 63, 64を備えている。この4段のD型フリップフロップ61～64は、本発明による4段の遅延回路に相当する。

## 【0086】

また、1段目のD型フリップフロップ61の出力データを-1倍する第1の乗算器65と、2段目のD型フリップフロップ62の出力データを9倍する第2の乗算器66と、3段目のD型フリップフロップ63の出力データを9倍する第3の乗算器67と、4段目のD型フリップフロップ64の出力データを-1倍する第4の乗算器68とを備えている。さらに、上記第1～第4の乗算器65～68の出力データを全て加算する加算器69, 70, 71を備えている。

## 【0087】

加算器71から出力されたデータは、ANDゲート72の一方の入力端に入力される。ANDゲート72の他方の入力端には、インバータ73を通過した反転クロックCKが入力される。また、初段のD型フリップフロップ61に入力される離散データは、上述の各回路ブロック61～71の遅延に合わせた量だけ遅延

させる遅延回路74を介してANDゲート75の一方の入力端に入力される。ANDゲート75の他方の入力端には、クロックCKが入力される。

### 【0088】

上記2つのANDゲート72, 75の出力データは、ORゲート76を介して出力される。これにより、クロックCKが“H”の期間中は、入力される離散データがそのままORゲート76より出力され、クロックCKが“L”の期間中は、上記式(8)に示すような演算により得られる補間データがORゲート76より出力される。

### 【0089】

なお、この図9は上記式(8)のような演算を行うための1つの回路例であり、これに限定されるものではない。例えば、図9に示した乗加算回路65～71の代わりに、以下のような構成を用いても良い。すなわち、1段目のD型フリップフロップ61の出力データと4段目のD型フリップフロップ64の出力データとを加算する第1の加算器と、2段目のD型フリップフロップ62の出力データと3段目のD型フリップフロップ63の出力データとを加算する第2の加算器と、上記第1の加算器の出力データを-1倍する第1の乗算器と、上記第2の加算器の出力データを9倍する第2の乗算器と、上記第1の乗算器の出力データと上記第2の乗算器の出力データとを加算する第3の加算器とにより乗加算回路を構成する。このようにすれば、乗算器を削減することができる。

### 【0090】

#### (第2の実施形態)

次に、本発明の第2の実施形態について説明する。第2の実施形態では、図12に示すデジタル基本関数を1クロックずつずらして加算（平均化）した関数を使用する。

### 【0091】

図10は、第2の実施形態によるデータ補間の原理を説明するための図である。図10に示すデータ構造において、横軸のA,B,C,…は1クロック毎に順次入力される各サンプル点の離散データを示し、そのデータ値が本実施形態のデジタル基本関数に対する係数となる。また、縦軸のa,b,c,…は、デジタル基本関数によ

り加工された離散データの中心位置を示す。

#### 【0092】

図10に示すように、離散データAに対するデジタル基本関数は、最初のクロック位置より2クロック前を先頭として各関数値（-1, 0, 9, 16, 9, 0, -1）を1クロックずつずらして与えられる。また、次の離散データBに対するデジタル基本関数は、最初のクロック位置を先頭として各関数値を1クロックずつずらして与えられる。同様に、順次入力される各離散データC,D,E,F,…に対するデジタル基本関数は、更に2クロックずつ遅れたクロック位置をそれぞれ先頭として、各関数値を1クロックずつずらして与えられる。

#### 【0093】

このようなデータ構造によると、例えば離散データCの値に応じて加工されるデジタル基本関数の中心位置cは、第1の実施形態と異なり1つのデータ位置の上に重なる。そこで、第2の実施形態では、このクロック位置cのデータと、その前後のクロック位置のデータとを用いて畳み込み演算を行うことにより、1つの間引きデータCから2つの補間データC1,C2を求める。

#### 【0094】

クロック位置cを中心とする前後3行分のデータをそれぞれ、c-, c, c+とする  
と、

$$c_{-} = -A + 9B + 9C - D \quad \cdots (9)$$

$$c = 16C \quad \cdots (10)$$

$$c_{+} = -B + 9C + 9D - E \quad \cdots (11)$$

と表される。

#### 【0095】

上記式(9)(10)より、

$$(c_{-} + c) = (-A + 9B + 25C - D) \quad \cdots (12)$$

と表され、また、上記式(10)(11)より、

$$(c + c_{+}) = (-B + 25C + 9D - E) \quad \cdots (13)$$

と表される。

#### 【0096】

この式(12)(13)をもとに振幅の調整分を考慮すると、1つの離散データCは、2つの補間データC1, C2に置き換えることができ、それらは、

$$C1 = (-A + 9B + 25C - D) / 32 \quad \dots (14)$$

$$C2 = (-B + 25C + 9D - E) / 32 \quad \dots (15)$$

のように表される。

### 【0097】

また、離散データCについて式(14)(15)に示される関係が他の離散データD, E, F, …についても同様に成り立つ。したがって、次式に示すように、1つの離散データDは2つの補間データD1, D2に、1つの離散データEは2つの補間データE1, E2に、1つの離散データFは2つの補間データF1, F2にそれぞれ置き換えられる。

$$D \rightarrow D1 = (-B + 9C + 25D - E) / 32, D2 = (-C + 25D + 9E - F) / 32$$

$$E \rightarrow E1 = (-C + 9D + 25E - F) / 32, E2 = (-D + 25E + 9F - G) / 32$$

$$F \rightarrow F1 = (-D + 9E + 25F - G) / 32, F2 = (-E + 25F + 9G - H) / 32$$

.....

### 【0098】

以上のように、あるサンプル点の離散データ（例えばC）に関して補間値を求める際には、その離散データを25倍した値と、1つ前のサンプル点の離散データを9倍した値と、2つ前のサンプル点の離散データを-1倍した値と、1つ後のサンプル点の離散データを-1倍した値とを加算し、その加算値を8で割ることによって第1の補間値（例えばC1）を得る。

### 【0099】

また、対象の離散データを25倍した値と、1つ前のサンプル点の離散データを-1倍した値と、1つ後のサンプル点の離散データを9倍した値と、2つ後のサンプル点の離散データを-1倍した値とを加算し、その加算値を8で割ることによって第2の補間値（例えばC2）を得る。

これを各サンプル点について行うことにより、元データを2倍にオーバーサンプルする。

### 【0100】

また、2倍にオーバーサンプルされた補間データをもとに同様の補間処理を行うことにより、元データを4倍にオーバーサンプルすることが可能である。さらに、4倍にオーバーサンプルされた補間データをもとに同様の補間処理を行うことにより、元データを8倍にオーバーサンプルすることが可能である。理論的には、同様の補間処理をn回繰り返し行うことにより、元データを $2^n$ 倍にオーバーサンプルすることが可能である。

#### 【0101】

図11は、第2の実施形態によるデータ補間装置の構成例を示す図である。図11に示すデータ補間装置は、2倍のオーバーサンプリングを行うための回路を3段カスケード接続した構成になっており、1段目～3段目の回路構成を符号の添数字(-1～-3)により区別している。また、添数字のみが異なり主符号が同じ回路構成は互いに同一の機能を有するものであることを示している。

#### 【0102】

1段目のオーバーサンプリング回路は、基準周波数(例えば44.1KHz)のクロックCKに従って動作する。5段のD型フリップフロップ81<sub>-1</sub>, 82<sub>-1</sub>, 83<sub>-1</sub>, 84<sub>-1</sub>, 85<sub>-1</sub>は、順次入力される離散データ(例えば16ビット)を基準周波数の1クロックCKずつ順に遅延させる。これらのD型フリップフロップ81<sub>-1</sub>～85<sub>-1</sub>は、請求項13による5段の遅延回路に相当する。

#### 【0103】

1段目のD型フリップフロップ81<sub>-1</sub>の出力タップから取り出されたデータは、-1倍の乗算器86<sub>-1</sub>(請求項14の第1の乗算器に相当)に入力される。2段目のD型フリップフロップ82<sub>-1</sub>の出力タップから取り出されたデータは、+9倍の乗算器87<sub>-1</sub>(請求項14の第2の乗算器に相当)と-1倍の乗算器93<sub>-1</sub>(請求項14の第5の乗算器に相当)とに入力される。

#### 【0104】

3段目のD型フリップフロップ83<sub>-1</sub>の出力タップから取り出されたデータは、25倍の乗算器88<sub>-1</sub>(請求項14の第3の乗算器に相当)に入力される。4段目のD型フリップフロップ84<sub>-1</sub>の出力タップから取り出されたデータは、-1倍の乗算器89<sub>-1</sub>(請求項14の第4の乗算器に相当)と+9倍の乗算器94<sub>-1</sub>

$-1$  (請求項14の第6の乗算器に相当) とに入力される。5段目のD型フリップフロップ $85_{-1}$ の出力タップから取り出されたデータは、 $-1$ 倍の乗算器 $95_{-1}$  (請求項14の第7の乗算器に相当) に入力される。

#### 【0105】

上記 $-1$ 倍乗算器 $86_{-1}$ ,  $9$ 倍乗算器 $87_{-1}$ ,  $25$ 倍乗算器 $88_{-1}$ および $-1$ 倍乗算器 $89_{-1}$ の出力データは、3つの加算器 $90_{-1}$ ,  $91_{-1}$ ,  $92_{-1}$ によって全て加算される。上記 $-1$ 倍乗算器 $93_{-1}$ ,  $25$ 倍乗算器 $88_{-1}$ ,  $9$ 倍乗算器 $94_{-1}$ および $-1$ 倍乗算器 $95_{-1}$ の出力データは、3つの加算器 $96_{-1}$ ,  $97_{-1}$ ,  $98_{-1}$ によって全て加算される。

#### 【0106】

加算器 $92_{-1}$ から出力されたデータは、ANDゲート $99_{-1}$ の一方の入力端に入力される。ANDゲート $99_{-1}$ の他方の入力端には、クロックCKが入力される。また、加算器 $98_{-1}$ から出力されたデータは、ANDゲート $100_{-1}$ の一方の入力端に入力される。ANDゲート $100_{-1}$ の他方の入力端には、インバータ $101_{-1}$ を通過した反転クロックCKが入力される。

#### 【0107】

上記2つのANDゲート $99_{-1}$ ,  $100_{-1}$ の出力データは、ORゲート $102_{-1}$ を介して出力される。これにより、クロックCKが“H”の期間中は、上記式(14)に示すような演算結果がORゲート $102_{-1}$ より出力され、クロックCKが“L”の期間中は、上記式(15)に示すような演算結果がORゲート $102_{-1}$ より出力される。

#### 【0108】

以上のような回路に対して、図10に示すような離散データA, B, C, …を順に入力することにより、ORゲート $102_{-1}$ からは、A, B, C1, C2, D1, D2, …のように2倍にオーバーサンプリングされた補間データが出力される。なお、式(14)(15)ではデータ値が全体として $1/32$ 倍されているが、そのための回路が1段目のオーバーサンプリング回路には設けられていない。図11の回路では、3段分をまとめて最終段の乗算器 $104$ で $1/32768$  ( $1/32^3$ ) 倍している。

#### 【0109】

上記ORゲート102-1の出力データは、2段目のオーバーサンプリング回路のD型フリップフロップ81-2に入力され、2段目の2倍オーバーサンプリングが実行される。2段目のオーバーサンプリング回路は、基準周波数の2倍の周波数のクロック2CKに従って動作すること以外は、1段目のオーバーサンプリング回路と同様である。

#### 【0110】

2段目のオーバーサンプリング回路の最終段に設けられたORゲート102-2の出力データは、3段目のオーバーサンプリング回路のD型フリップフロップ81-3に入力され、3段目の2倍オーバーサンプリングが実行される。3段目のオーバーサンプリング回路は、基準周波数の4倍の周波数のクロック4CKに従って動作すること以外は、1段目のオーバーサンプリング回路と同様である。

#### 【0111】

3段目のオーバーサンプリング回路の最終段に設けられたORゲート102-3の出力データは、8倍周波数のクロック8CKに従ってD型フリップフロップ103に保持された後、1/512倍乗算器104を介して出力される。

#### 【0112】

この図11のようにデータ補間装置を構成した場合も、ユニットパルスのデータを入力した場合に得られる標本化関数は、図3④に示したものとほぼ同じものとなる。したがって、第2の実施形態によるデータ補間処理を適用した場合も、離散データ間の値を1回以上微分可能な関数を用いてより滑らかに補間することが可能となる。これにより、ローパスフィルタを不要とし、その位相特性の劣化を防止することができる。

#### 【0113】

また、ある1つの離散データに関して補間値を求める場合には、限られた数（上記式(14)(15)で示されるように4個）の離散データの値のみを考慮すればよいことになり、処理量を大幅に削減することができる。しかも、有限台の範囲外の各離散データについては、本来考慮すべきであるが処理量や精度等を考慮して無視しているというわけではなく、理論的に考慮する必要がないため、打ち切り誤差の発生を防止することもできる。

### 【0114】

また、第2の実施形態では、図10に示すように、図12に示すデジタル基本関数を1クロックずつずらして加算（平均化）した関数を使用して補間データを求めており、この関数内で既に平均化演算は行われている。したがって、上記式(14)(15)による補間演算を行った後に平均化演算を行う必要がなく、平均化回路を省略することができる。

### 【0115】

なお、上記第1および第2の実施形態では、デジタル基本関数として、1クロック毎にデータ値を-1, 1, 8, 8, 1, -1と変化させたものを用いた。この数値が最良ではあるが、これ以外の数値を用いたもの（例えば、両脇に当たる部分のウェイトを-1ではなく1あるいは0にしたもの、真中に当たる部分のウェイトを8以外の値にしたものなど）をデジタル基本関数としても良い。この場合でも、`sinc`関数を用いる従来と比べれば、離散データ間を滑らかにつなぐ良好な補間データを簡単な演算によって得ることができる。

### 【0116】

以上に説明した第1および第2の実施形態によるデータ補間の手法は、ハードウェア構成、DSP、ソフトウェアの何れによっても実現することが可能である。例えばソフトウェアによって実現する場合、本実施形態のデータ補間装置は、実際にはコンピュータのCPUあるいはMPU、RAM、ROMなどで構成され、RAMやROMに記憶されたプログラムが動作することによって実現できる。

### 【0117】

したがって、コンピュータが上記本実施形態の機能を果たすように動作させるプログラムを例えばCD-ROMのような記録媒体に記録し、コンピュータに読み込ませることによって実現できるものである。上記プログラムを記録する記録媒体としては、CD-ROM以外に、フロッピーディスク、ハードディスク、磁気テープ、光ディスク、光磁気ディスク、DVD、不揮発性メモリカード等を用いることができる。また、上記プログラムをインターネット等のネットワークを介してコンピュータにダウンロードすることによっても実現できる。

### 【0118】

また、コンピュータが供給されたプログラムを実行することにより上述の実施形態の機能が実現されるだけでなく、そのプログラムがコンピュータにおいて稼働しているOS（オペレーティングシステム）あるいは他のアプリケーションソフト等と共同して上述の実施形態の機能が実現される場合や、供給されたプログラムの処理の全てあるいは一部がコンピュータの機能拡張ボードや機能拡張ユニットにより行われて上述の実施形態の機能が実現される場合も、かかるプログラムは本発明の実施形態に含まれる。

#### 【0119】

図1および図11に示すデータ補間装置は、複数の離散データを順次入力することによってデータ補間を行うものであるが、単一のデジタルデータを入力することによって、図3④のような標本化関数を生成する標本化関数生成装置としても用いることが可能である。

また、上記のデータ補間装置は、例えば、離散的なデジタルデータを連続的なアナログ信号に変換するデジタルーアナログ変換や、圧縮時に離散化されたデータから補間によって元のデータを復元する伸長処理、画像データの拡大処理などに応用することが可能である。

#### 【0120】

その他、上記各実施形態は、本発明を実施するにあたっての具体化の一例を示したものに過ぎず、これらによって本発明の技術的範囲が限定的に解釈されてしまうものである。すなわち、本発明はその精神、またはその主要な特徴から逸脱することなく、様々な形で実施することができる。

#### 【0121】

##### 【発明の効果】

本発明は上述したように、順次入力される離散データについて、注目するサンプル点の離散データとその周辺数個のサンプル点の離散データとに対して、デジタル基本関数の値に応じた重み付け加算をすることにより補間データを求めるようにしたので、全域において有限回微分可能な有限台の標本化関数を極めて簡単な四則演算のみで得ることができ、当該標本化関数に基づく滑らかなデータ補間を簡単な演算で行うことができる。したがって、ローパスフィルタの位相歪みや

打ち切り誤差のない良好な補間データを得るために演算時間を短縮化することができるとともに、その演算回路を簡素化することができる。

【図面の簡単な説明】

【図 1】

第 1 の実施形態によるデータ補間装置の構成例を示す図である。

【図 2】

第 1 の実施形態によるデータ補間の原理を説明するための図である。

【図 3】

図 1 に示したデータ補間装置にユニットパルスのデータを入力した場合に得られるオーバーサンプル結果を示す特性図である。

【図 4】

図 3 の④に示す波形関数と、図 1 3 に示す波形関数とを比較のために並べて示す特性図である。

【図 5】

図 4 に示した標本化関数の一部拡大図である。

【図 6】

図 4 に示した標本化関数の周波数特性を示す図である。

【図 7】

第 1 の実施形態によるデータ補間装置の他の構成例を示す図である。

【図 8】

第 1 の実施形態によるデータ補間装置の他の構成例を示す図である。

【図 9】

第 1 の実施形態によるデータ補間装置の他の構成例を示す図である。

【図 10】

第 2 の実施形態によるデータ補間の原理を説明するための図である。

【図 11】

第 2 の実施形態によるデータ補間装置の構成例を示す図である。

【図 12】

本実施形態で用いるデジタル基本関数を示す図である。

**【図13】**

図12に示すデジタル基本関数から生成される従来の標本化関数を示す特性図である。

**【図14】**

s i n c 関数の説明図である。

**【図15】**

図14に示した標本化関数を用いたデータ補間の説明図である。

**【符号の説明】**

1 -1, 2 -1, 3 -1 D型フリップフロップ

4 -1 -1 倍乗算器

5 -1, 6 -1 ANDゲート

7 -1 インバータ

8 -1 ORゲート

9 -1 8倍乗算器

10 -1 -1 倍乗算器

11 -1, 12 -1 ANDゲート

13 -1 インバータ

14 -1 ORゲート

15 -1, 16 -1 加算器

17 -1 D型フリップフロップ

18 -1 加算器

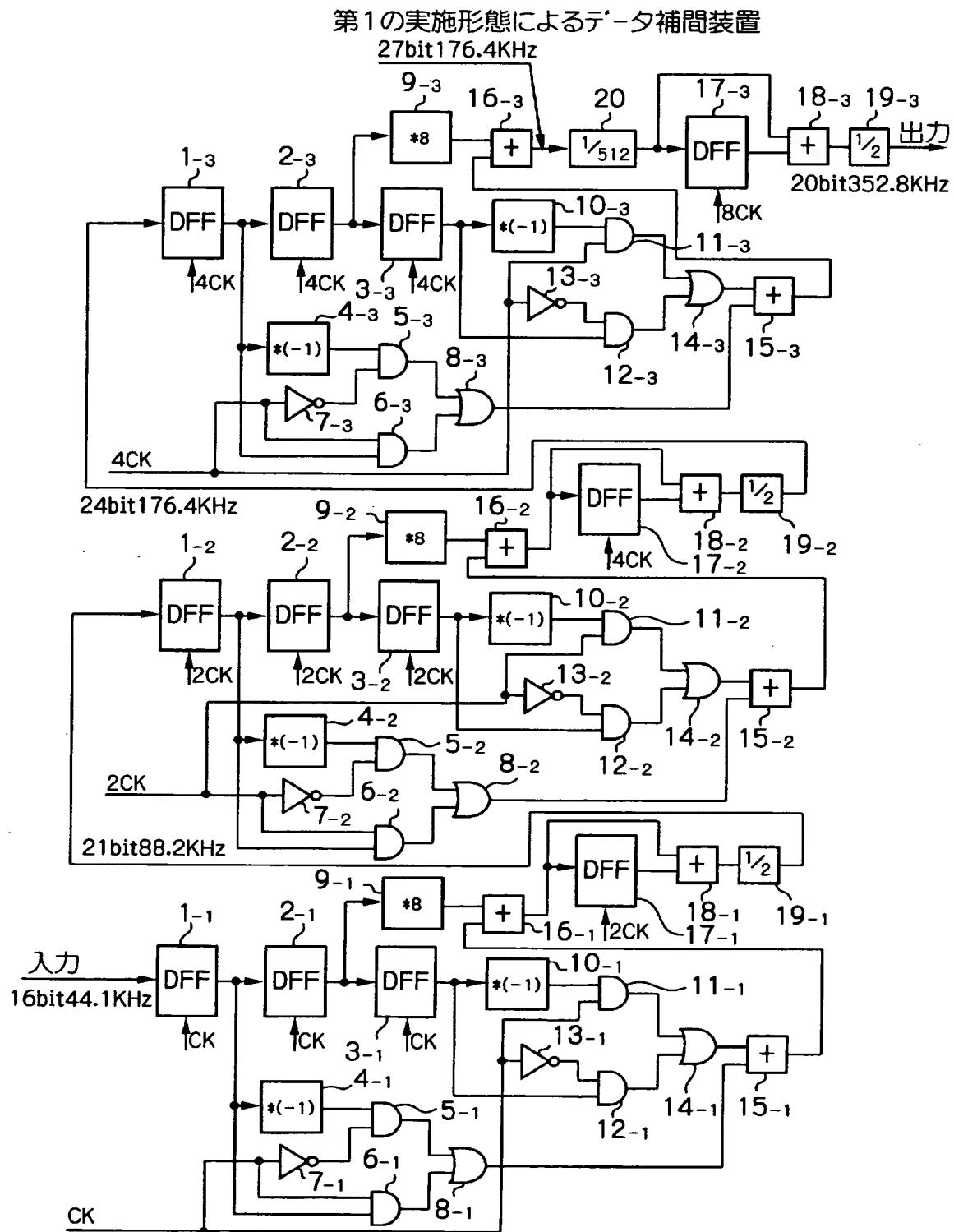
19 -1 1/2倍乗算器

20 1/512倍乗算器

【書類名】

図面

【図 1】



【図2】

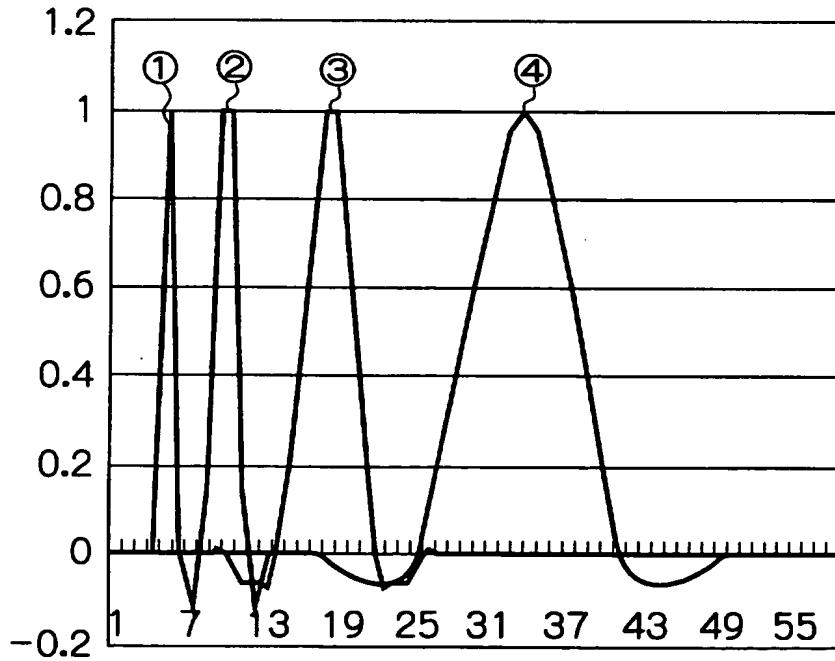
## 第1の実施形態によるデータ補間演算のデータ構造

	A	B	C	D	E	F	...
a	-1	1					
b	8	-1	1				
c	8	1					
d	1	8	-1				
e	-1	8	1				
f		1	8	-1			

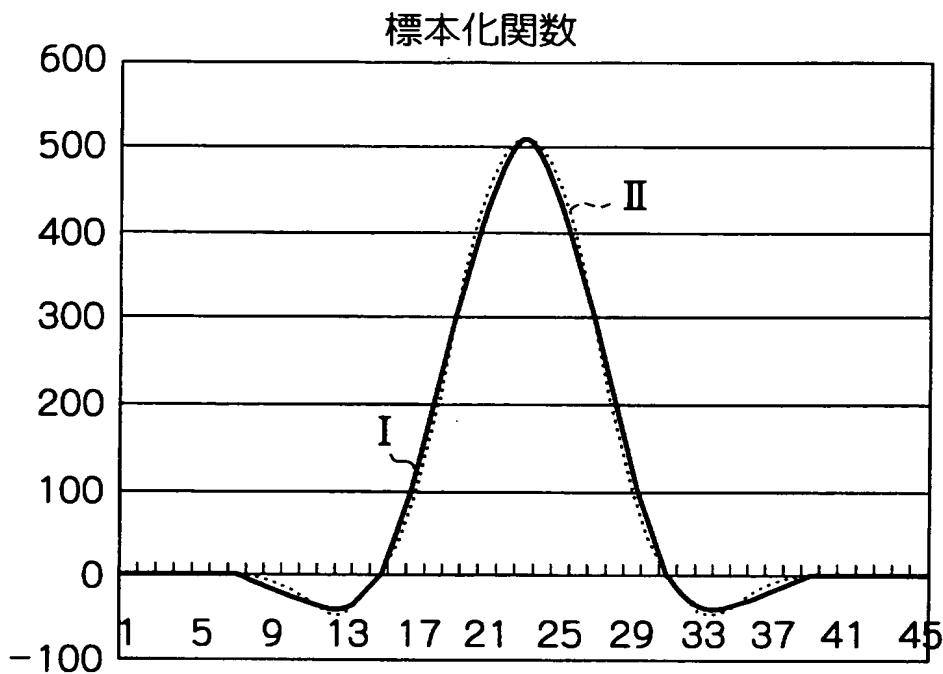
⋮

【図3】

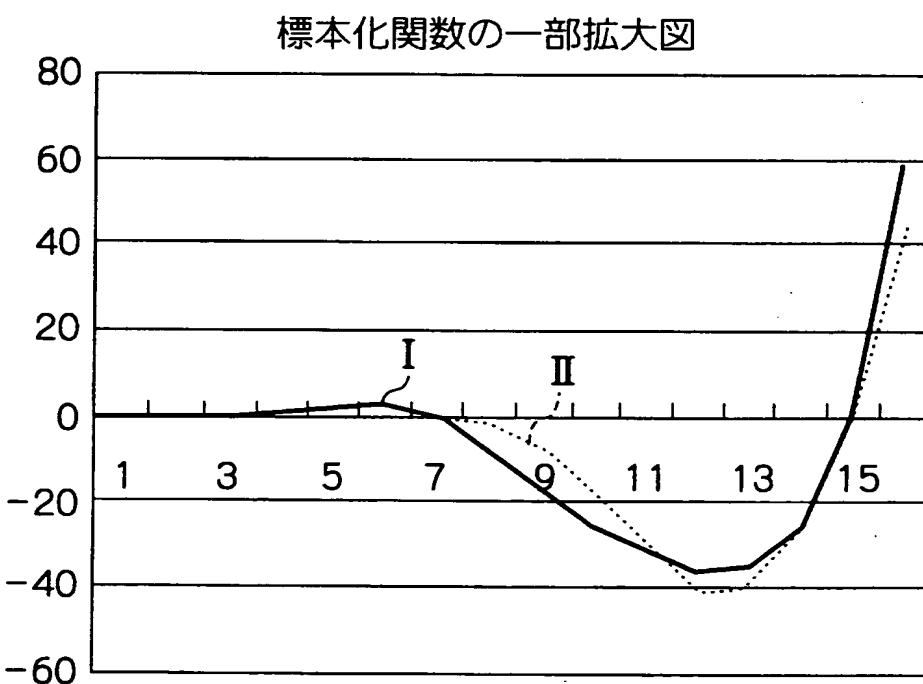
## 各段階のオーバーサンプル結果



【図4】

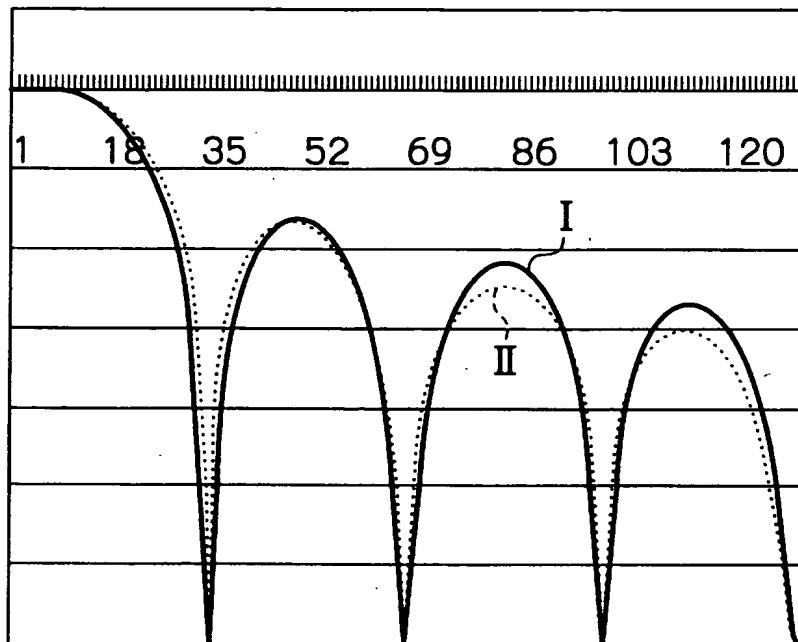


【図5】



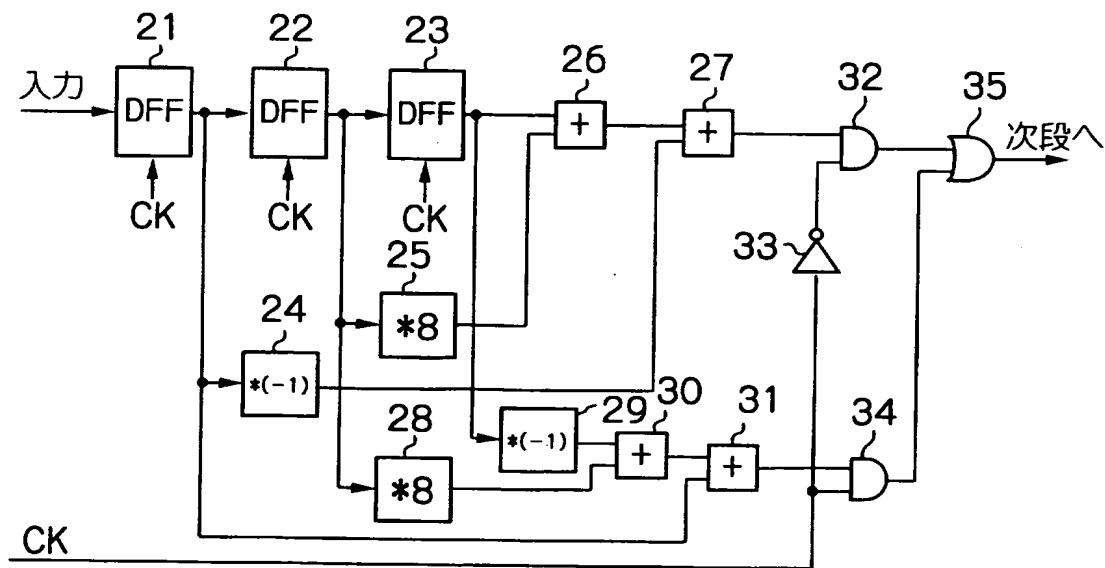
【図 6】

## 周波数特性



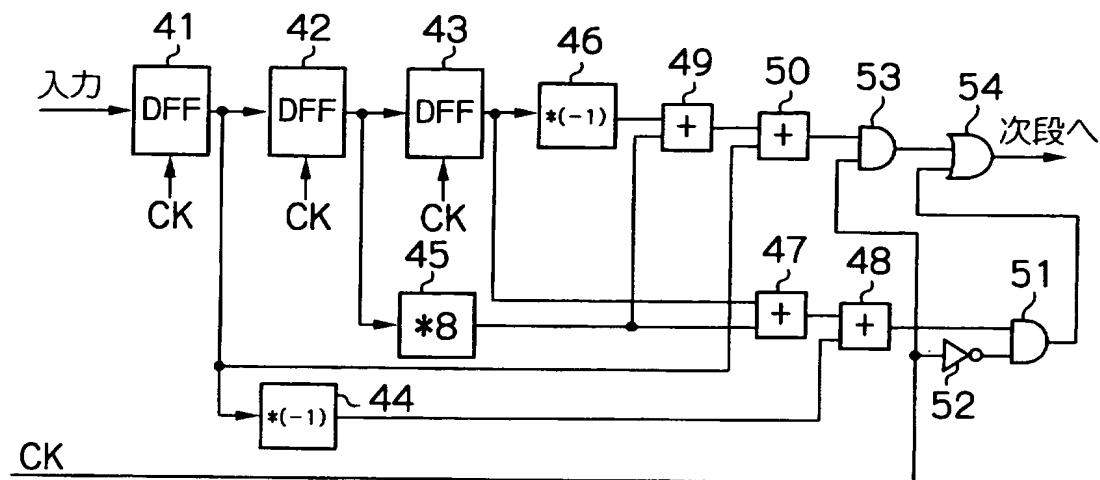
【図 7】

## データ補間装置の他の構成例



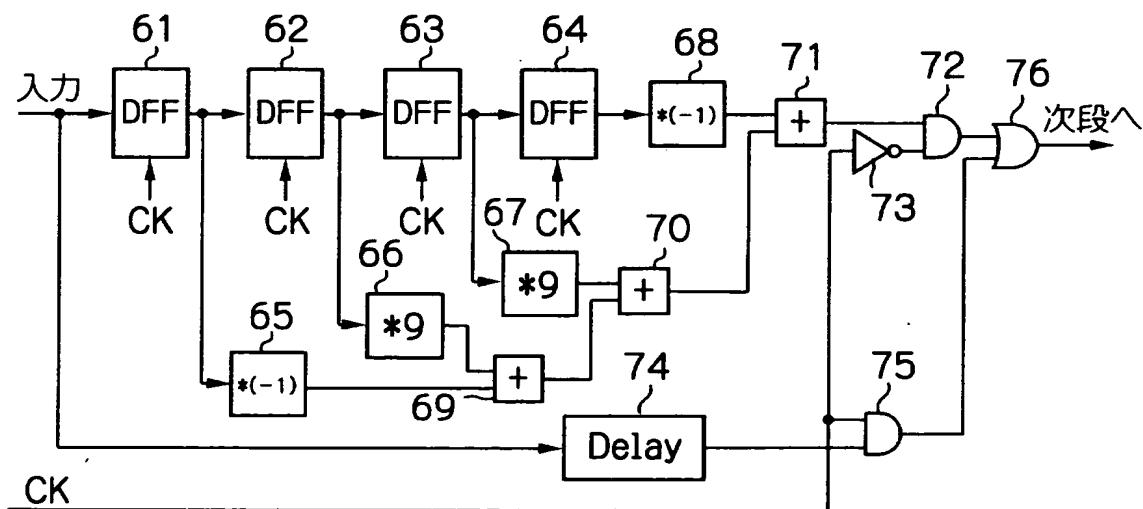
【図8】

データ補間装置の他の構成例



【図9】

データ補間装置の他の構成例



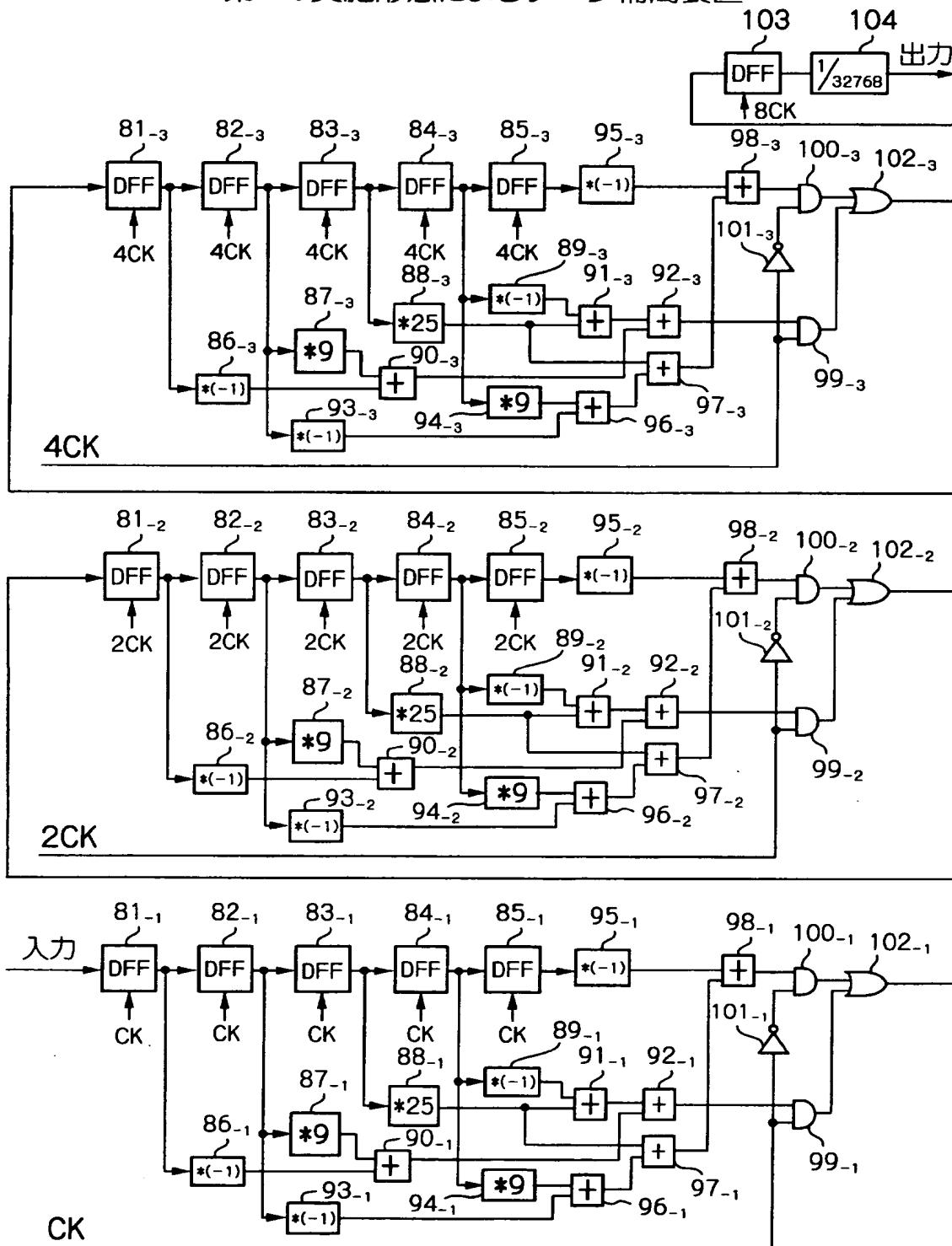
【四一〇】

#### 第2の実施形態によるデータ補間演算のデータ構造

	A	B	C	D	E	F	...
a	9	-1					
	-16	0					
b	9	9	-1				
	0	-16	0				
c	1	9	9	-1			
	-1	0	-16	0			
d		-1	9	9	-1		
		0	-16	0			
e			-1	9	9	-1	
			0	-16	0		
f				-1	9	9	
				0	-16		
				-1	9	9	
				0	-1	9	
					0	-1	

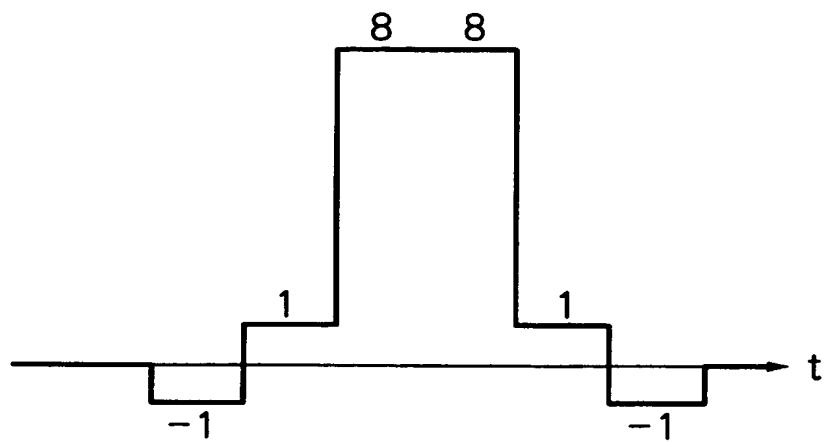
【図 11】

## 第2の実施形態によるデータ補間装置



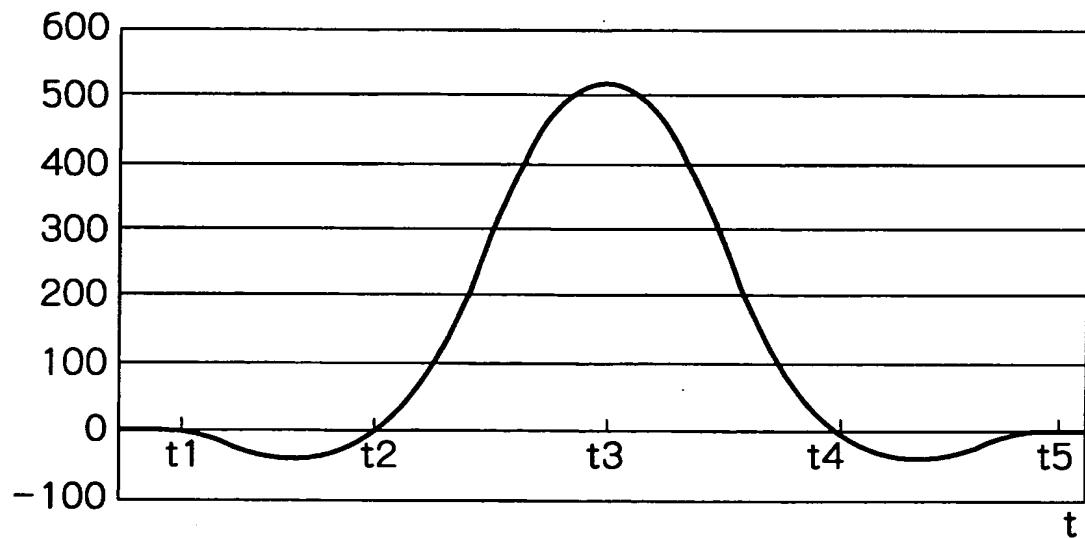
【図12】

## デジタル基本関数

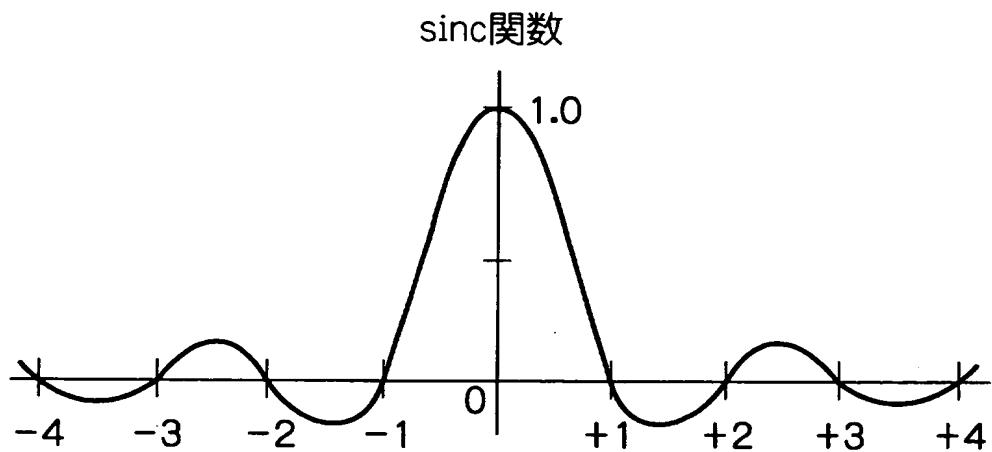


【図13】

## 有限台の標本化関数

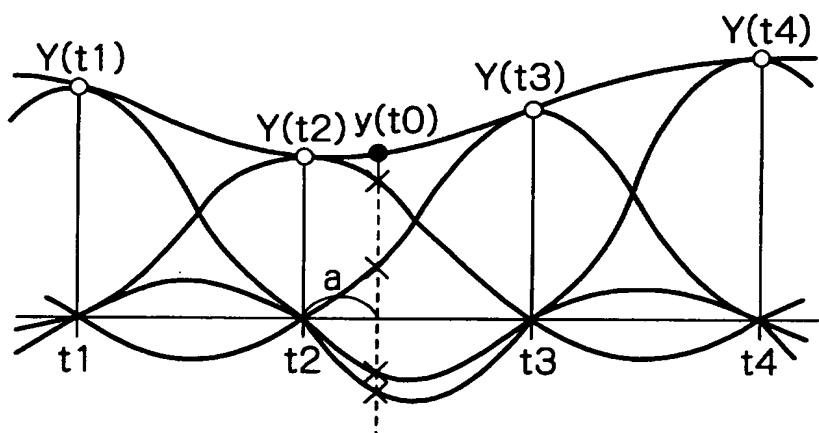


【図 1.4】



【図 1.5】

データ補間の説明図



【書類名】 要約書

【要約】

【課題】 全域において1回以上微分可能な有限台の標本化関数に基づく滑らかなデータ補間をより簡単な演算で行うことができるようとする。

【解決手段】 順次入力される離散データを遅延させる数段の遅延回路 $1_{-1}, 2_{-1}, 3_{-1}$ と、数段の遅延回路 $1_{-1}, 2_{-1}, 3_{-1}$ の各出力段から取り出されるそれぞれのデータに対して、デジタル基本関数 $(-1, 1, 8, 8, 1, -1)$ の値に応じた重み付け加算をすることによって補間データを求める乗加算回路 $4_{-1} \sim 16_{-1}$ とを設け、全域において1回以上微分可能な有限台の標本化関数を極めて簡単な四則演算のみで得ることができるようにし、ローパスフィルタの位相歪みや打ち切り誤差のない良好な補間データを得るための演算時間を短縮化することができるようになるとともに、その演算回路を簡素化できるようとする。

【選択図】 図1

【書類名】 出願人名義変更届

【提出日】 平成15年 5月 9日

【あて先】 特許庁長官殿

【事件の表示】

【出願番号】 特願2001-173883

【承継人】

【識別番号】 503084059

【氏名又は名称】 有限会社ニューロソリューション

【承継人代理人】

【識別番号】 100105784

【弁理士】

【氏名又は名称】 橋 和之

【電話番号】 049-249-5122

【手数料の表示】

【予納台帳番号】 070162

【納付金額】 4,200円

【提出物件の目録】

【物件名】 承継人であることを証明する書面 1

【援用の表示】 平成11年特許願第173245号の出願人名義変更届  
に添付のものを援用する。

【物件名】 委任状 1

【援用の表示】 平成11年特許願第173245号の出願人名義変更届  
に添付のものを援用する。

【プルーフの要否】 要

## 認定・付加情報

特許出願の番号	特願 2001-173883
受付番号	50300771432
書類名	出願人名義変更届
担当官	古田島 千恵子 7288
作成日	平成 15 年 9 月 8 日

## &lt;認定情報・付加情報&gt;

## 【承継人】

【識別番号】	503084059
【住所又は居所】	東京都世田谷区中町五丁目 4 番 7 号
【氏名又は名称】	有限会社ニューロソリューション
【承継人代理人】	申請人
【識別番号】	100105784
【住所又は居所】	埼玉県川越市下新河岸 87 番地 64 橘国際特許 事務所
【氏名又は名称】	橘 和之

特願2001-173883

出願人履歴情報

識別番号 [595016543]

1. 変更年月日 1999年 5月24日  
[変更理由] 住所変更  
住 所 埼玉県浦和市中尾409-1-D115  
氏 名 酒井 康江

2. 変更年月日 2002年 3月 6日  
[変更理由] 住所変更  
住 所 埼玉県さいたま市中尾409-1-D115  
氏 名 酒井 康江

特願2001-173883

出願人履歴情報

識別番号 [503084059]

1. 変更年月日 2003年 3月 3日

[変更理由] 新規登録

住所 東京都世田谷区中町五丁目4番7号  
氏名 有限会社ニューロソリューション